01 Sep 2008

Implementation of Neuroidentifiers Trained by PSO on a PLC Platform for a Multimachine Power System

Curtis Alan Parrott

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

2008 IEEE Swarm Intelligence Symposium
St. Louis MO USA, September 21-23, 2008

Implementation of Neuroidentifiers Trained by PSO on a PLC Platform for a Multimachine Power System

Curtis Parrott, *Student Member, IEEE*, and Ganesh K. Venayagamoorthy, *Senior Member, IEEE*

Real-Time Power and Intelligent Systems Laboratory
Missouri University of Science and Technology, Rolla, MO 65409, USA
cap9qd@mst.edu and *gkumar@ieee.org*

***Abstract* – Power systems are nonlinear with fast changing dynamics. In order to design a nonlinear adaptive controller for damping power system oscillations, it becomes necessary to identify the dynamics of the system. This paper demonstrates the implementation of a neural network based system identifier, referred to as a neuroidentifier, on a programmable logic controller (PLC) platform. Two separate neuroidentifiers are trained using the particle swarm optimization (PSO) algorithm to identify the dynamics in a two-area four machine power system, one neuroidentifier for Area 1 and the other for Area 2. The power system is simulated in real time on the Real Time Digital Simulator (RTDS). The PLC implementing two neural networks and the PSO training algorithm is interfaced in a real time to the RTDS. Typical results are presented showing that PLC platform is able to implement the neuroidentifiers to sufficiently identify the dynamics of the two-area four machine power system.**

I. INTRODUCTION

Power systems are nonlinear with fast changing dynamics. In order to design a nonlinear adaptive controller for damping power system oscillations, it becomes necessary to identify the dynamics of the system. Neural networks have been shown to identify dynamics of systems [9]. Power system adaptive controllers including power system stabilizers, automatic voltage regulators and governors have been designed consisting of two neural networks, one for a model and the other for a controller [4, 2]. Several training algorithms have been reported for neural networks. Backpropagation and particle swarm optimization (PSO) are known to be suitable for online/quasi-online training [2, 5].

Digital power system control has been explored on a wide variety of platforms, including digital signal processors (DSPs) and field programmable logic arrays (FPGAs), in great detail. However, this research has neglected the staple of industrial control, the programmable logic controller. The programmable logic controller (PLC) platform is used extensively in industry due to its very high reliability and expandability.

Manuscript received June 15, 2008. This work was supported in part by the National Science Foundation Career Award ECCS #0348221 - "Scalable Learning and Adaptation with Intelligent Techniques and Neural Networks for Reconfiguration and Survivability of Complex Systems" and U.S. Department of Education GAANN Fellowship - "Advanced Computational Techniques and Real-Time Simulation Studies for the Next Generation Energy Systems"

C. A. Parrott is with the Missouri University of Science and Technology, Rolla, MO 65401 USA; cap9qd@mst.edu

G. K. Venayagamoorthy is with the Missouri University of Science and Technology, Rolla, MO 65401 USA; *gkumar@ieee.org*

This expandability includes a wide variety of digital and analog I/O modules along with many different communication modules. The PLC is also designed with a powerful processor with the ability to do real-time control of a wide variety of control application [1]. To the knowledge of the authors, computational intelligence techniques including neural networks and PSO have not been implemented on PLCs which are known to be robust platforms for industrial control applications.

The RTDS is a custom parallel processing hardware platform that allows power systems to be simulated and its accessories (controllers, transformers, relays) to be tested in real-time [2]. Through the use of analog I/O, power control devices can be seamlessly tested as if they were part of the physical power system running on the simulator. This allows for the testing of any such control device containing low voltage I/O and allows the gauging of this control scheme as a legitimate real-world application. The ability of the RTDS for control and protection system testing has been further explored in [2]. This makes the PLC - RTDS platform, an ideal platform for testing the viability of the PLC as a real-world control platform for computational intelligence techniques, including for neuroidentification of generator speed deviations. In this paper, a study for implementing neuroidentifiers based on neural networks trained with PSO for predicting generator speed oscillations is investigated on a PLC platform for the purpose of damping power system oscillations. The PLC platform implements two neural networks required to realize identification of the speed deviations of two generators in a two-area four machine power system [7]. These neural networks are trained using the PSO algorithm [5]. In addition, a real-time interface is illustrated between the RTDS hardware and the PLC for implementing system identifiers using neural networks and PSO.

II. MULTIMACHINE TWO-AREA POWER SYSTEM

The multimachine power system studied to demonstrate the PLC implementation of a neuroidentifier is the standard two-area four machine power system in Fig. 1 [7].

This power system consists of two fully symmetrical areas linked together by two transmission lines. Each area is equipped with two synchronous generators rated at 20 kV/900 MVA. All the generators are equipped with identical speed governors and turbines, AVRs and exciters. Generators G1 and G3 are both also equipped with CPSSs during the testing of fault conditions. The layout of both generators G1 and G3 can be seen in Fig.2. The loads for

978-1-4244-2705-5/08/$25.00 ©2008 IEEE

each area are represented as constant impedances and are split between the two areas such that Area 1 transmits approximately 413 MW of power to Area 2. Three electromechanical modes of oscillation are present in this system: two inter-plant/intra-area modes, one in each area, and one inter-area low-frequency mode [8]. The parameters of this system are given in [8].

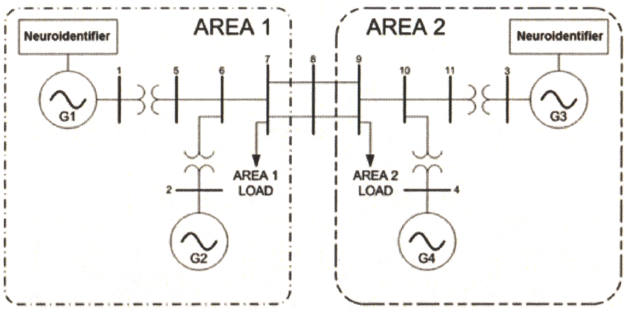

Fig. 1. Multi-Machine Two-area power system.

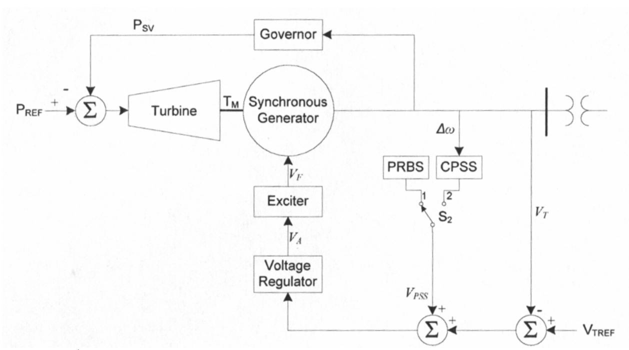

Fig. 2. Generator G1/G3 control arrangement during neuroidentifier (NI) training and NI testing.

This power system is a test system commonly used to show the effectiveness of controllers in damping slow-mode oscillations [7,8]. This system is implemented in RTDS such that the practical implementation of an neuroidentifier on the PLC platform can be demonstrated in real-time system. Although the system is interfaced to the RTDS simulator, the simulations are run in real-time and very closely approximate real-world implementations. This allows the PLC platform and developed neuroidentifier to be evaluated as a practical, real world, identification system as compared to a pure non-real time simulation study.

III. Design of a Neuroidentifier

The design of the neuroidentifier (NI) is partly based on the form used in implementing an indirect adaptive neurocontroller (IDNC). The layout of this IDNC consists of two separate neural networks: a neurocontroller and a neuroidentifier. This work implements the neuroidentifier for system identification and in future work a neurocontroller will be implemented and trained using this neuroidentifier network. The diagram for the NI training can be seen in Fig. 3. The training of the NI is carried out using the PSO algorithm, which is discussed later in this section. The dashed lines in Fig. 3 represent this update to the respected neural network.

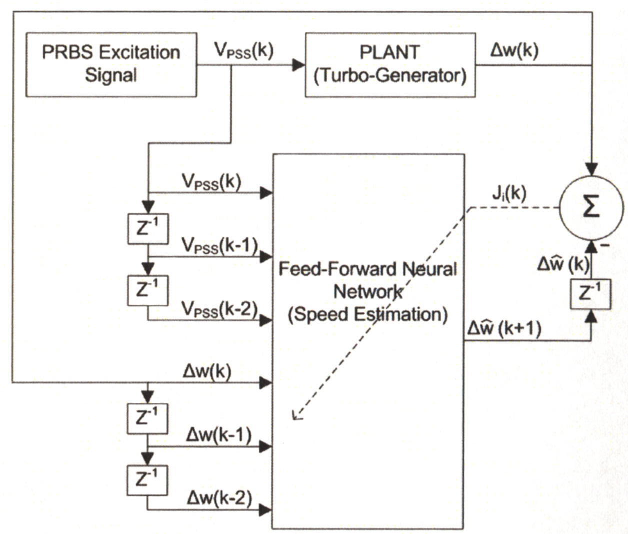

Fig. 3. Neuroidentifier structure during training.

A. Neuroidentifier

The neuroidentifier is used to estimate the speed deviations of a generator in the next sample time step. This model is developed using the series-parallel nonlinear auto regressive moving average model [9]. The model output at the time step $k+1$ depends on both past n values of its output as well as past m values of its input. The inputs and outputs of the model are speed deviation of the plant (generator G1 or G3) and the output of the CPSS, and the estimated speed deviations respectively. Here both n and m are chosen to be 2. The main reason for choosing three time step values is because a third order system is sufficient for the modeling the generator dynamics for this study. The model is a multi-layered feedforward neural network.

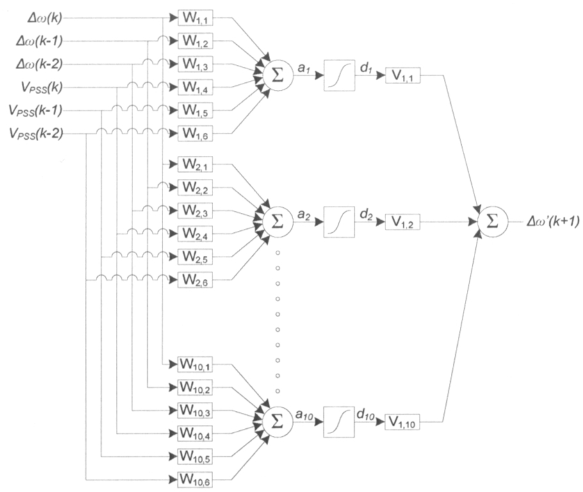

Fig. 4. Detailed neuroidentifier structure.

$$\overline{X} = \begin{bmatrix} \Delta\omega(k), \Delta\omega(k-1), \Delta\omega(k-2), \\ V_{PSS}(k), V_{PSS}(k-1), V_{PSS}(k-2) \end{bmatrix} \quad (1)$$

$$a_i = \sum_{j=1}^{10} W_{i,j} \cdot X_j \quad (2)$$

$$d_i = \frac{1}{1+e^{-1 \cdot a_i}} \quad (3)$$

$$\Delta\omega(k+1) = \sum_{i=1}^{10} V_i \cdot d_i \quad (4)$$

B. PSO Algorithm

PSO is a type of evolutionary computing technique. The algorithm is based on the simulation of the social interaction of birds within a flock and schools of fish. Being a population based search algorithm, a swarm consists of particles, each of which are a potential solution to the problem to be solved or optimized. The changes in the particles position in the search space is influenced by the past knowledge of the swarm as well as the particles own past knowledge of the search space.

At algorithm initialization each particle is randomly initialized to a point in the search space, as well as given a random starting velocity. The particle is then flown through the search space with the initial velocity. The particle is then evaluated as to how well it solves the problem at hand; this evaluation is called the particle's fitness. This is then compared to the particles memory of its best solution of the problem, the *pbest* position. If the newest solution is better than the current *pbest* (the current fitness lower than the *pbest* fitness), the *pbest* position is updated to the current position. Once all the particles have been evaluated the *pbest* with the lowest fitness is compared to the *gbest* position fitness. If this *pbest*'s value is lower than the current *gbest* fitness then the *gbest* position is update to this *pbest*'s location. This *gbest* represents the social aspect of the algorithm. After these updates have been done the PSO equations are again evaluated, and take the form seen in (5) and (6). Also, an example of a single particle update can be seen in figure 6 where: $x_{id}(k)$ is the i^{th} particle's d^{th} dimension current position; $x(k+1)$ is the i^{th} particle's d^{th} dimension position after the PSO update, at the next time step; $v_{id}(k)$ is the i^{th} particle's d^{th} dimension current velocity; $v_{id}(k+1)$ is the i^{th} particle's d^{th} dimension velocity at the next time step; p_{id} is the *pbest* position of i^{th} particle's; p_{gd} is the groups best position or *gbest* for the d^{th} dimension; w is the inertial weight constant; c_1 is the cognition acceleration constant; and c_2 is the social acceleration constant.[5]

$$\begin{aligned} v_{id}(k+1) = w \cdot v_{id}(k) + c_1 \cdot rand_1 \cdot (p_{id}(k) - x_{id}(k)) \\ + c_2 \cdot rand_2 \cdot (p_{gd}(k) - x_{id}(k)) \end{aligned} \quad (5)$$

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k+1) \quad (6)$$

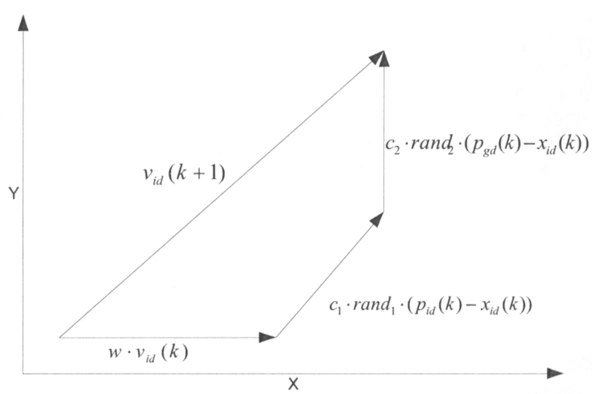

Fig. 5. PSO particle update process for two dimensional case. [5]

IV. PROGRAMMABLE LOGIC CONTROLLER AND REAL-TIME SIMULATION PLATFORM

The PLC platform is a tried and true platform for control and automation. Due to its design, it has many advantages over general purpose computer, digital signal processor, and field programmable logic array based control systems. The PLC is design to be in an industrial environment and is built to withstand this environment which can contain electrical noise, electromechanical interference, mechanical vibrations, extreme temperatures above 140 degrees Fahrenheit and non-condensing humidity of 95% [10]. These other platform afore mentioned would require modification to withstand these kind of environmental conditions. PLCs are also highly modular and only require a simple module change to add extra features while a complete system redesign would be needed with a computer, DSP or FPGA based design. Since the PLC executes a single program in a sequential fashion it can recover from power failure quickly since there is no boot-up procedure, and thus have a larger edge against the computer systems [10].

The PLC platform used in this research is the Allen-Bradley ControlLogix 5561 processor along with component rack, power supply and analog IO cards. This line of PLC processors and hardware provide the needed processing power to execute the control algorithms in question.

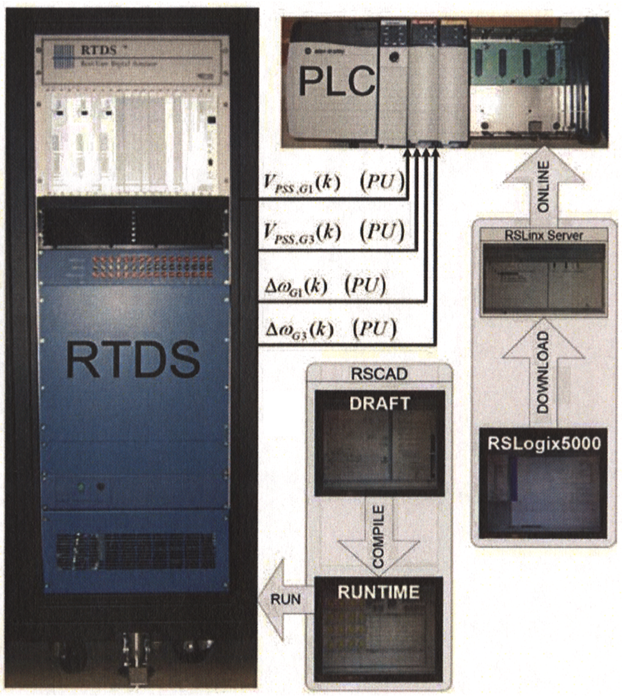

Fig. 6. RTDS and PLC platform monitor signal connections, RTDS and PLC programming connections.

As mentioned before, the control system is interfaced to the RTDS. This simulator allows for the simulation of power systems in real-time while connecting auxiliary control components to the simulation via analog I/O. The RTDS and PLC hardware test setup for this study is shown in Fig. 4.

V. Neuroidentifier and PSO Implementation

The neuroidentifier was trained via off-line PSO algorithm due to the computational complexity of the fitness evaluation of PSO. All of the control components, including the model and PSO where implemented in structured text PLC programming language using Allen-Bradley's RSLogix 5000 programming software.

In order to train the networks the speed deviation of each turbo generator is calculated and transmitted to the PLC via the analog channels of the RTDS; these channels transmit a voltage from -10 volts to 10 volts DC. This speed deviation is also up scaled in the RTDS hardware to take full advantage of the 16 bits of resolution available in the analog channels and then later downscaled back to the original value when received by the PLC. This is done in order to minimize the quantization error of the analog channels to maximize the resolution of the transmitted signal. The PRBS forced training signal is also transmitted to the PLC in a similar fashion.

The training of the neural network model was accomplished by implementing the PSO algorithm in the PLC. First the PLC would capture 25 seconds of speed deviation and PRBS signal at 40 Hz. Then the PSO particles are initialized randomly between [-0.1,0.1]. Next each of the PSO solutions are evaluated. This is done by applying the testing data points captured earlier to the neural network and calculating the mean-squared-error (MSE) between the identifier output and the speed deviation at the next time step. The neural network model's fitness equation takes the form given in (7) and (8). This MSE over all input samples is the fitness for each particle and is used to update the *pbest* and *gbest* values. Once a satisfactory solution for the NI is attained the training is halted and testing can commence.

$$J_i(k) = \Delta\hat{\omega}_{G1}(k) - \Delta\omega_{G1}(k) \tag{7}$$

$$fitness_i = \frac{1}{1000}\left[\sum_{k=0}^{999} J_i(k)^2\right] \tag{8}$$

VI. Implementation Results

Experiments have been done with the RTDS to prove the PLC capable of running the neuroidentifier and PSO training algorithm. The following figures illustrate the results found. The training of the neuroidentifier was done under operating condition 1, as seen in table 1. To show the quality of the neuroidentifier solution three operating points where used, along with two different fault conditions. In the first condition the pseudo-random binary sequence is applied to the input of the exciter, causing the system to oscillate. An example of this signal is illustrated in figure 13. The results for these forced oscillation tests are illustrated in figures 7, 9 and 11. The next fault condition used to demonstrate the quality of neuroidentifier solution is a 3-phase 10-cycle fault on bus 8. During this tests an individual CPSS is connected to both generators G1 and G3, as well as its output V_{PSS} being input to the neuroidentifier, as illustrated in figure 3 and 6. The prediction results are illustrated in figures 8, 10 and 12. In all of these figures the green curve is the neuroidentifier prediction of the speed deviation and the blue curve is the actual speed deviation, both in PU.

TABLE I. LOADS FOR EACH OF THE THREE OPERATING POINTS.[11]

Operating Point	Power Transfer	Load Area 1	Load Area 2
1st	246MW	1120MW	1180MW
2nd	446MW	920MW	1380MW
3rd	490MW	967MW	1767MW

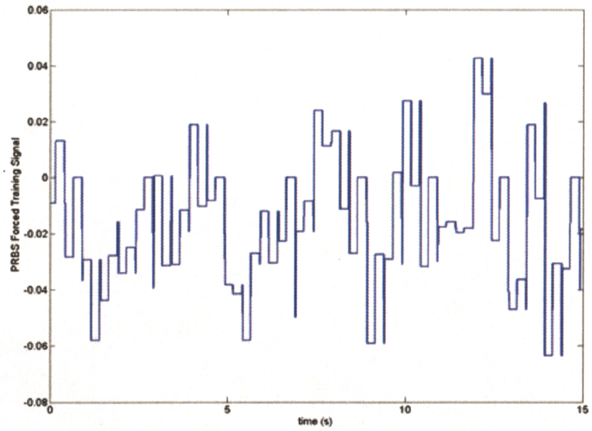

Fig. 7. PRBS signal applied to exciter to force generator speed deviation.

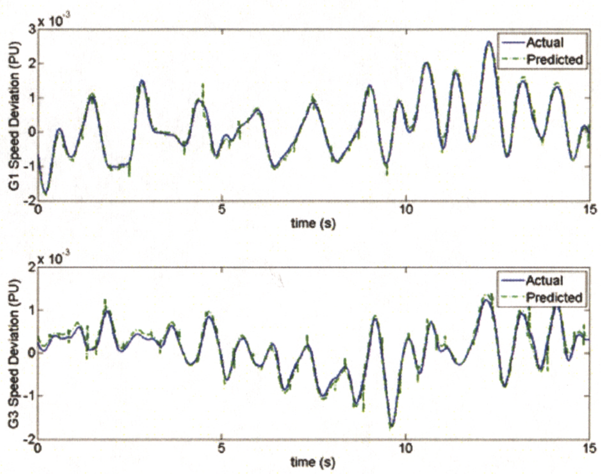

Fig. 8. Generator G1 and G3 speed deviation and neuroidentifier speed prediction. This shows the forced response due to the PRBS signal under load operating point 1.

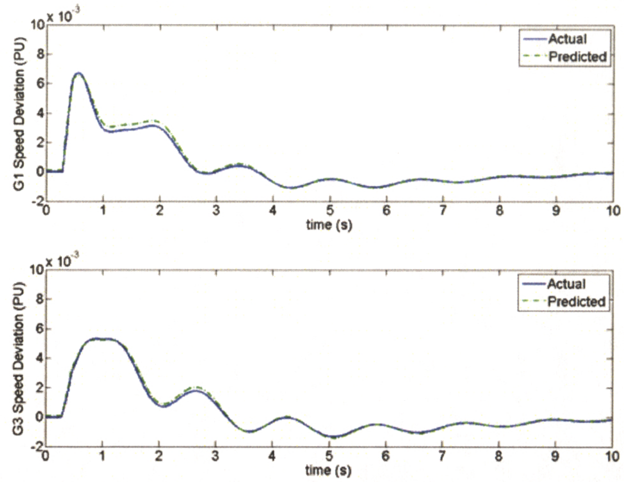

Fig. 9. Generator G1 and G3 speed deviation and neuroidentifier speed prediction. This shows the forced response due to a 3-phase 10-cycle fault on bus 8 under load operating point 1.

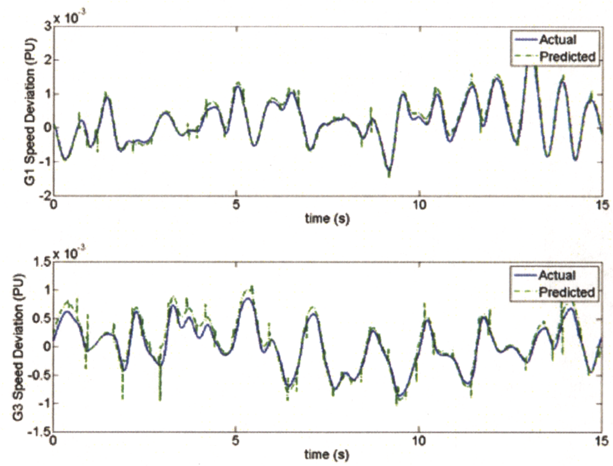

Fig. 10. Generator G1 and G3 speed deviation and neuroidentifier speed prediction. This shows the forced response due to the PRBS signal under load operating point 2.

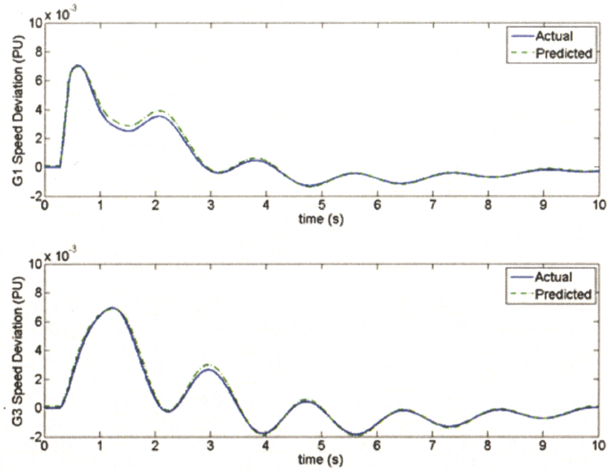

Fig. 11. Generator G1 and G3 speed deviation and neuroidentifier speed prediction. This shows the forced response due to a 3-phase 10-cycle fault on bus 8 under load operating point 2.

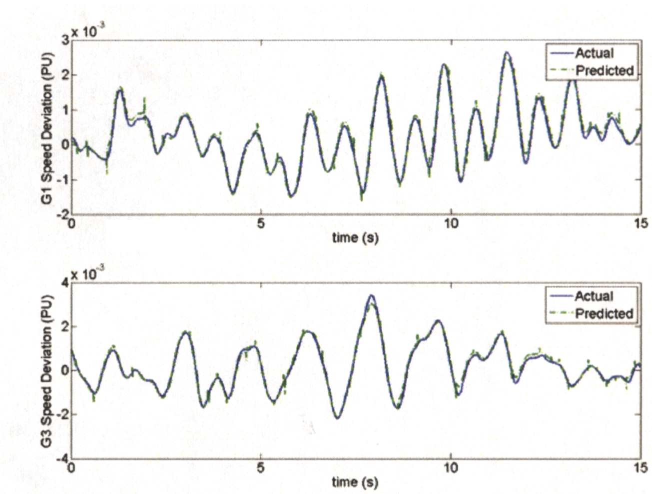

Fig. 12. Generator G1 and G3 speed deviation and neuroidentifier speed prediction. This shows the forced response due to the PRBS signal under load operating point 3.

VII. Conclusion

Two neuroidentifiers for predicting speed deviations of two generators in a two-area four machine power system has been implemented on a programmable logic platform. The training algorithm for the neuroidentifiers, the particle swarm optimization, has been also implemented on the PLC platform. The real time implementation results obtained on the power system real time digital simulator and the PLC show that neuroidentifiers trained with PSO are able to sufficiently identify the speed deviations of the generators in the two different areas of the power system during forced and natural disturbances. This paper has shown that the PLC is a promising platform for implementing neural networks and particle swarm optimization algorithms. Future work involves the development of a controller on the PLC platform to damp power system oscillations.

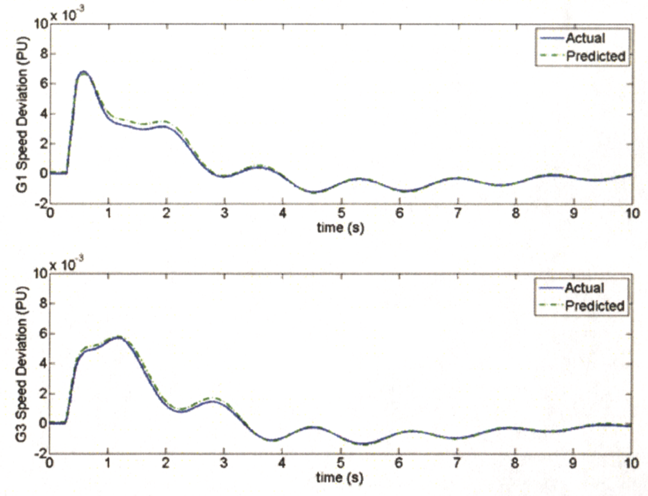

Fig. 13. Generator G1 and G3 speed deviation and neuroidentifier speed prediction. This shows the forced response due to a 3-phase 10-cycle fault on bus 8 under load operating point 2.

VIII. References

[1] M. Basler, and R. Schaefer, 'Understanding Power System Stability', *Pulp and Paper Industry Technical Conference,* 2007, pp. 37-47

[2] G. K. Venayagamoorthy, and R. G. Harley, "Two separate continually online-trained neurocontrollers for excitation and turbine control of turbogenerators", *IEEE Transactions on Industrial Applications*, vol. 38, no. 3, May/June 2002, pp887-893

[3] P. Forsyth, T. Maguire, and R. Kuffel, 'Real Time Simulation for Control and Protection System Testing', *IEEE Power Electronics Specialists Conference Proceedings*, 2004, pp. 329-335

[4] W. Lui, G. K. Venayagamoorthy, and D. Wunsch, 'Design of an adaptive neural network based power system stabilizer', Science Direct Neural Networks 16, 2003, pp. 891-898

[5] R. Kiran, S. R. Jetti and G. K. Venayagamoorthy, 'Online Training of a Generalized Neuron with Particle Swarm Optimization', International Joint Conference on Neural Networks, 2006

[6] K. T. Erickson, 'Programmable Logic Controllers', *IEEE Potentials*, 1996

[7] P. Kundur, *Power system stability and control.* New York: McGraw-Hill, 1994

[8] M. Klein, G. Rogers, and P. Kundur, 'A fundamental study of inter-area oscillations in power systems', *IEEE Transactions on Power Systems*, 1991, pp. 914-921

[9] K. Narendra, 'Identification and control of dynamical system using neural networks', *IEEE Transactions on Neural Networks*, 1990, 4-27

[10] K. Erickson, *Programmable Logic Controllers: An Emphasis on Design and Applications*, Rolla: Dogwood Valley Press, 2005, pp. 1-20

[11] S. Ray, and G. K. Venayagamoorthy, 'Wide-Area Signal-Based Optimal Neurocontroller for a UPFC, *IEEE Transactions on Power Delivery*, 2008